



Performances analysis of a Hybridizable discontinuous Galerkin solver for the 3D Helmholtz equations in geophysical context

Marie Bonnasse-Gahot, Henri Calandra, Julien Diaz, Stéphane Lanteri

► To cite this version:

Marie Bonnasse-Gahot, Henri Calandra, Julien Diaz, Stéphane Lanteri. Performances analysis of a Hybridizable discontinuous Galerkin solver for the 3D Helmholtz equations in geophysical context. Third EAGE workshop on High performance computing for Upstream, Oct 2017, Athens, Greece. hal-01677290

HAL Id: hal-01677290

<https://hal.science/hal-01677290>

Submitted on 15 Jan 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Performances analysis of a Hybridizable discontinuous Galerkin solver for the 3D Helmholtz equations in geophysical context

M. Bonnasse-Gahot¹, H. Calandra², J. Diaz¹ and S. Lanteri³

¹ INRIA-Bordeaux Sud-Ouest, Team Magique-3D, Pau, France.

² TOTAL E&P Research and Technology USA.

³ INRIA-Sophia-Antipolis-Méditerranée, Team Nachos, Sophia-Antipolis, France.

Introduction

For several decades, the Oil and Gas industry has been committed to produce more and more hydrocarbons in response to the growing world demand for energy. Always seeking deeper and further, exploration and development has become economically challenging as a result of increased geological and above ground complexity, stronger environmental constraints and pressure on costs.

Significantly enhanced computational algorithms and more powerful computers have provided a much better understanding of the distribution and description of complex geological structures, opening new frontiers to unexplored geological areas as well as helping limiting the risks and overall costs of deep and ultra-deep offshore drilling.

The progress of multi component seismic data acquisition and the fast evolution of new technologies in the rock physics labs provide the opportunity to develop new families of algorithms which include more complex physics. Modern imaging techniques such as Full Wave Form Inversion (FWI) rely on intensive usage of full 3D physical Wave Equation engines, so that accurate and scalable numerical methods are required to solve efficiently wave equation kernels.

The FWI method defines an iterative procedure based on an inversion process. Its main steps are, starting from an initial velocity model, **a)** to compute the solution of the wave equation for the N sources of the seismic acquisition campaign; **b)** to evaluate, for each source, a residual defined as the difference between the wavefields recorded at receivers on the top of the subsurface during the acquisition campaign and the numerical wavefields; **c)** to compute the solution of the wave equation using the residuals as sources; **d)** to update the velocity model by cross correlation of images produced at steps **a)** and **c)**. Finally, the different steps **a)** to **d)** are repeated until convergence of the velocity model is achieved.

We then have to solve $2N$ wave equations at each iteration. The number of sources, N , is usually large (about 1000) and the efficiency of the inverse solver is thus directly related to the efficiency of the numerical method used to solve the wave equation. The solution can be performed in the time domain or in the frequency domain regime and we focus here on the second setting. The drawback of time domain is that it requires to store the solution at each time step of the forward simulation. The difficulties related to frequency domain inversion lie in the solution of huge linear systems, which cannot be achieved today when considering realistic 3D elastic media, even with the progress of high-performance computing. In this context, the goal is to develop

new forward solvers that reduce the number of degrees of freedom without hampering the accuracy of the numerical solution. We show here how hybridizable discontinuous Galerkin (HDG) methods can be employed to solve the elastic equations. We provide scalability results in order to show that HDG are very well adapted to high performance computing. We show that they outperform classical DG methods and we analyze the influence of the solver on the scalability in 3D.

Hybridizable Discontinuous Galerkin method

We consider discontinuous Galerkin (DG) methods formulated on fully unstructured meshes, which are more convenient than finite difference methods on Cartesian grids to handle the topography of the subsurface. Moreover, DG methods are more adapted than continuous Galerkin (CG) methods to deal with *hp*-adaptivity. This last characteristics is crucial to adapt the mesh to the different regions of the subsurface which is generally highly heterogeneous. Nevertheless, the main drawback of classical DG methods is that they are expensive because they require a large number of degrees of freedom as compared to CG methods on a given mesh (see Figs. 1-2).

We have chosen to consider a new class of DG method, the hybridizable DG (HDG) method (see Kirby et al. (2012) for more details). They have been introduced by B. Cockburn, J. Gopalakrishnan and R. Lazarov in 2009 for the resolution of second order elliptic problems (see Cockburn et al. (2009)). Instead of solving a linear system involving the degrees of freedom of all volumic cells of the mesh, the principle of HDG consists in introducing a Lagrange multiplier representing the trace of the numerical solution on each face of the mesh. Hence, it reduces the number of unknowns of the global linear systems and the volumic solution is recovered thanks to a local computation on each element (see Fig. 3). The HDG methods have been considered in some recent works, for example, for the solution of the elastodynamic equations in time domain Nguyen et al. (2011a) and for Maxwell's equations S. Lanteri and Perrussel (2013); Nguyen et al. (2011b).

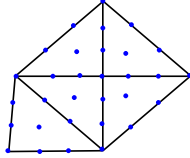


Figure 1: Distribution of the global degrees of freedom (dof) for the FEM with an interpolation order of 3

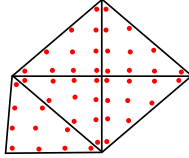


Figure 2: Distribution of the global dof for the DG method with an interpolation order of 3

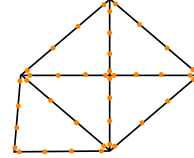


Figure 3: Distribution of the global dof for the HDG method with an interpolation order of 3

We consider the first order formulation of the 3D elastic wave equations in frequency domain. We have, for $\mathbf{x} = (x, y, z) \in \Omega \subset \mathbb{R}^3$

$$\begin{cases} \iota \omega \rho(\mathbf{x}) \mathbf{v}(\mathbf{x}) &= \nabla \cdot \underline{\underline{\sigma}}(\mathbf{x}) + f(\mathbf{x}) & \text{in } \Omega, \\ \iota \omega \underline{\underline{\sigma}}(\mathbf{x}) &= \underline{\underline{C}}(\mathbf{x}) \underline{\underline{\varepsilon}}(\mathbf{v}(\mathbf{x})) & \text{in } \Omega, \end{cases} \quad (1)$$

where ι is the imaginary unit, ω the angular frequency; $\rho(\mathbf{x})$ defines the mass density and $f(\mathbf{x})$ the source term which is generally associated to volumic forces. The vector $\mathbf{v}(\mathbf{x}) = (v_x(\mathbf{x}), v_y(\mathbf{x}), v_z(\mathbf{x}))^T$ is the velocity vector and $\underline{\underline{\varepsilon}}$ the strain tensor, where $\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial j} + \frac{\partial v_j}{\partial i} \right)$, $i, j = x, y, z$ and $\underline{\underline{\sigma}}$ is the stress tensor. In the general case, $\sigma_{ij} = \sum_{k=\{x,y,z\}} \sum_{l=\{x,y,z\}} C_{ijkl} \varepsilon_{kl}$. The tensor $\underline{\underline{C}}$ is a fourth order symmetric tensor containing the elastic coefficients. It possesses sym-

metry properties, $C_{ijkl} = C_{jikl} = C_{ijlk} = C_{klij}$ and it is positive for all symmetric tensors ξ :

$$\sum_{i,j,k,l=\{x,y,z\}} C_{ijkl} \xi_{ij} \xi_{kl} \geq \alpha \sum_{i,j=\{x,y,z\}} \xi_{ij}^2.$$
 For simplicity, we do not describe the boundary conditions.

The main difference between Discontinuous Galerkin Methods and classical Finite Element Method lies in the fact that the basis functions are only assumed piecewise continuous. Assuming that physical parameters are piecewise constant (or polynomial) we approximate \mathbf{v} and $\underline{\underline{\sigma}}$ on each element K of a triangulation (or mesh) \mathcal{T}_h of Ω by polynomial functions.

To obtain the HDG formulation of the elastic waves equations, we write the weak variational formulation of system (1) into a linear system

$$\mathbb{A}^K \underline{W}^K + \mathbb{C}^K \underline{\Lambda} = F^K, \quad (2)$$

where \underline{W}^K is a vector containing the unknowns associated to degrees of freedom of \mathbf{v} and $\underline{\underline{\sigma}}$ belonging to the element K , while $\underline{\Lambda}$ is a vector containing all the unknowns associated to the Lagrange multipliers which represents the numerical trace of the solution. In our case, $\underline{\Lambda}$ represents the numerical trace of \mathbf{v} on each face. As the matrix \mathbb{A}^K is invertible, we express the initial unknowns of the problem, $(\mathbf{v}, \underline{\underline{\sigma}})$, as functions of the Lagrange multiplier $\underline{\Lambda}$

$$\underline{W}^K = (\mathbb{A}^K)^{-1} (F^K - \mathbb{C}^K \underline{\Lambda}).$$

To obtain the global linear system, we need to introduce a second equation. Thanks to the physics properties, this last equation is easily deduced by enforcing the continuity of the normal stress over each face of the mesh. It can be written as a linear system

$$\mathbb{B} \underline{W} + \mathbb{L} \underline{\Lambda} = 0, \quad (3)$$

Replacing \underline{W} in this last equation, we obtain the global linear system in $\underline{\Lambda}$ that we have to solve

$$\mathbb{A}^{HDG} \underline{\Lambda} = F^{HDG}. \quad (4)$$

Once this system is solved, the unknowns \mathbf{v}_h and $\underline{\underline{\sigma}}_h$ can be computed locally, element by element, by solving (2).

It is worth noting that, except the solution of the global linear system, all the steps of the HDG algorithm are easily parallelizable.

Numerical Results

First, we have compared the performances of the 2D HDG method with those of classical 2D nodal DG methods like the Internal Penalty Discontinuous Galerkin (IPDG) method (Rivière (2008)) on the Marmousi test-case. Using a direct solver (Mumps), we have shown that the memory consumption for the solution of the linear system, which is the main bottleneck of harmonic problems, is, for example, divided by 3 using HDG methods (see Fig. 4), and that the computational time of the linear solver is divided by 4 for a same interpolation order $p = 3$ (see Fig. 5).

Then, we have considered the 3D case and we have compared the HDG computational performances with the ones of a classical finite elements (FE) method for a same number of degrees of freedom (dof). We have shown that by increasing the number of dof, the resolution of the HDG linear system requires less memory than the resolution of the FE system. Moreover, we have been also able to reduce the memory consumption by proving that the HDG formulation for the elastic waves equations is a symmetric formulation. The HDG method for seismic imaging has been implemented as a proof of concept and is now being implemented into the FWI algorithm.

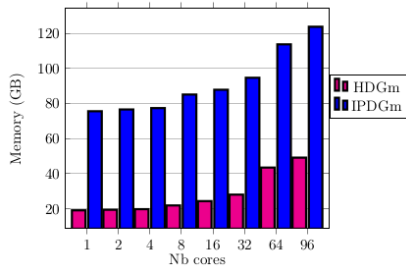


Figure 4: Memory consumption of classical DG and hybridizable DG.

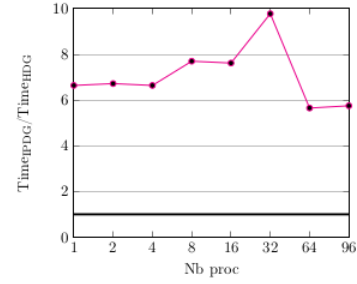


Figure 5: Speedup of hybridizable DG vs. classical DG

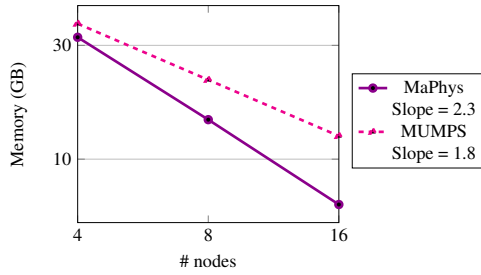


Figure 6: Average memory of the solver for one node.

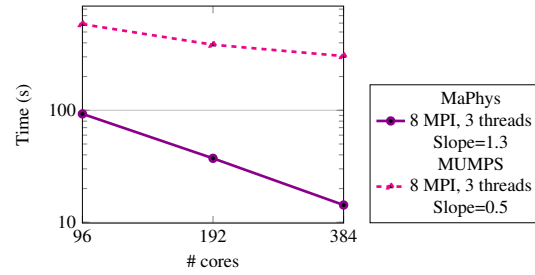


Figure 7: Execution time for the resolution of the HDG- \mathbb{P}_3 system.

However, despite the memory and computational gains with respect to classical methods, it is still not possible to tackle realistic 3D problems, even in a high performance computing framework. Therefore, we also have to focus on the solver part. We have coupled our method with an hybrid solver (Maphys) that combines direct and iterative solver using an algebraic domain decomposition method. We have shown that Maphys allows for a division of the memory consumption by 1.3 when compared to Mumps (see Fig. 6), without hampering the accuracy of the solution. The computational time (see Fig. 7) of the resolution is also improved by the use of the Maphys. The next step is the implementation of a multi-right hand side feature in Maphys, in order to handle the thousands of sources required by an inverse problem solver.

Acknowledgements

The authors acknowledge the support by the Inria-Total strategic action DIP (Depth Imaging Partnership, dip.inria.fr), the Brazilian and European project HPC4E (High Performance Computing for Energy, hpc4e.eu) and the European project GEAGAM (Geophysical Exploration using Advanced Galerkin Methods, sites.google.com/site/geagamnetwork/).

References

- Cockburn, B., Gopalakrishnan, J. and Lazarov, R. [2009] Unified hybridization of discontinuous Galerkin, mixed and continuous Galerkin methods for second order elliptic problems. *SIAM Journal Numerical Analysis*, (47), 1319–1365.
- Kirby, R., Sherwin, S. and Cockburn, B. [2012] To CG or to HDG: a comparative study. *J.Sci.Comput.*, (51), 183–212.
- Nguyen, N.C., Peraire, J. and Cockburn, B. [2011a] High-order implicit hybridizable discontinuous Galerkin methods for acoustics and elastodynamics. *Journal of Computational Physics*, (230), 3695–3718.
- Nguyen, N.C., Peraire, J. and Cockburn, B. [2011b] Hybridizable discontinuous Galerkin methods for the time-harmonic Maxwell’s equations. *Journal of Computational Physics*, (230 (19)), 7151–7175.

- Rivière, B. [2008] *Discontinuous Galerkin methods for solving elliptic and parabolic equations: theory and implementation*.
- S. Lanteri, L.L. and Perrussel, R. [2013] Numerical investigation of a high order hybridizable discontinuous Galerkin method for 2d time-harmonic Maxwell's equations. *Int.J.Comput.Math.Electr.Electron.Eng.*, (32), 1112–1138.